Multiresolution Adaptive Block-Coordinate Descent Methods for Imaging Problems

Edgar Desainte-Maréville[1], Marion Foare[1,3], Paulo Gonçalves[1], Nelly Pustelnik[2], Elisa Riccietti[1]

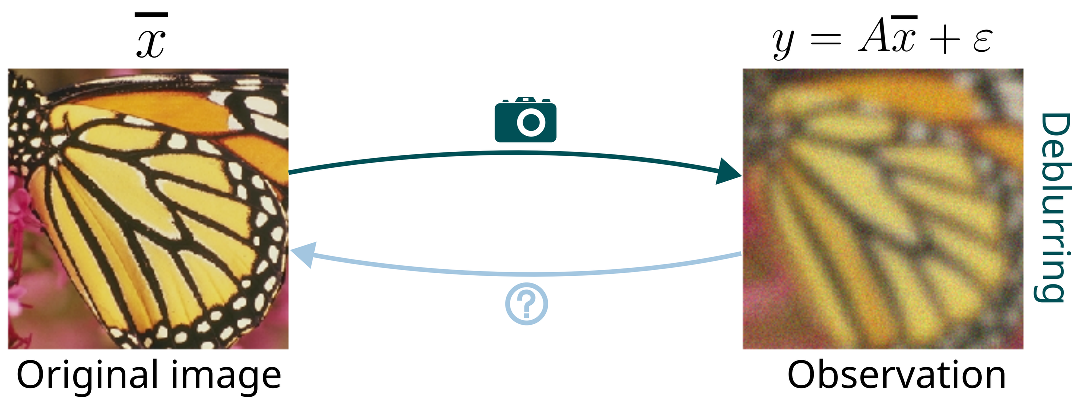
[1]Inria OCKHAM team, LIP, ENS Lyon [2]CNRS, Laboratoire de Physique, ENS Lyon [3]CPE Lyon

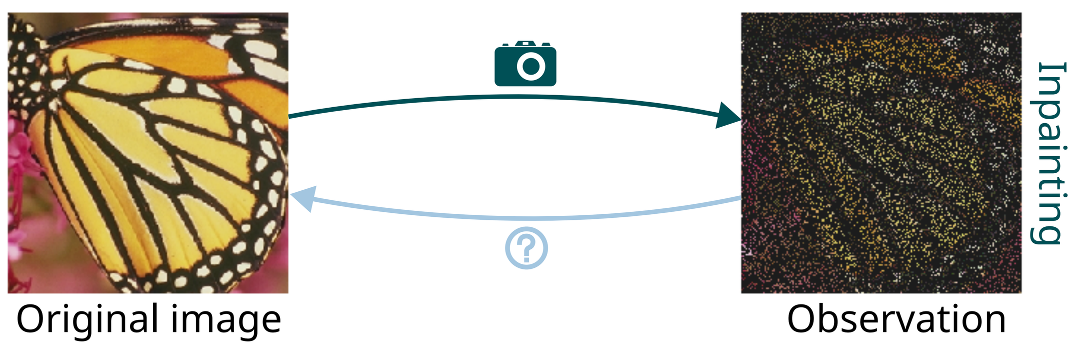
Work funded by the ANR MEPHISTO project (ANR-24-CE23-7039-01)

Motivation: How to adaptively select blocks in multiresolution block-coordinate descent algorithms.

Imaging inverse problems

Goal: Reconstruct an image from a degraded observation.





Ill-posed inverse problem

The inverse operator may *not exist* or lead to *irregular estimate:*

- Solution may *not be unique*
- Small variations in y ⟶ big variations in $\hat{x}$

Classical optimization method

Build a reconstruction by solving an optimization problem:

$$\hat{x} \in \underset{x \in \mathbb{R}^d}{\text{argmin}} \frac{1}{2}\|Ax - y\|_2^2 + \lambda R(x)$$

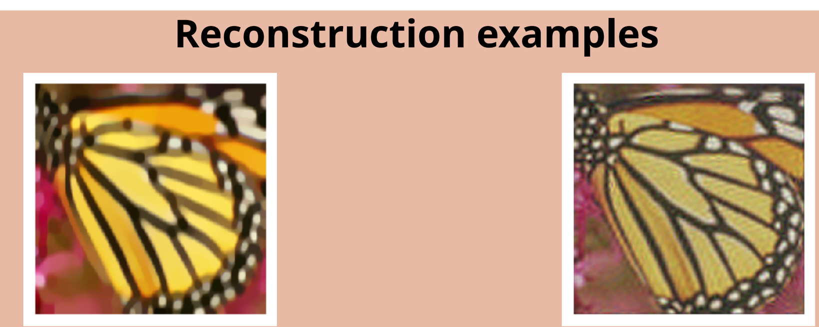
Data-fidelity — Ensures that $A\hat{x} \approx y$

Regularization — Reduces the search space by imposing some prior on the solution

The **Forward-Backward** algorithm

Given: Step size $\tau > 0$, regularization parameter λ
Input: Initial point x^0
for $k = 0, 1, 2, \ldots$ **do**
  $x^{k+\frac{1}{2}} = x^k - \tau A^\top(Ax^k - y)$
  $x^{k+1} = \text{prox}_{\tau\lambda R}(x^{k+\frac{1}{2}})$
end for

Reconstruction examples



$R(x) = TV(x) = \|\nabla x\|_1$ $R(x) = \|Wx\|_1$

Multiresolution decomposition

Wavelet transform: multiscale representation that decomposes an image into its low-frequency part (*approximation coefficients*) and its high-frequency components (*detail coefficients*).

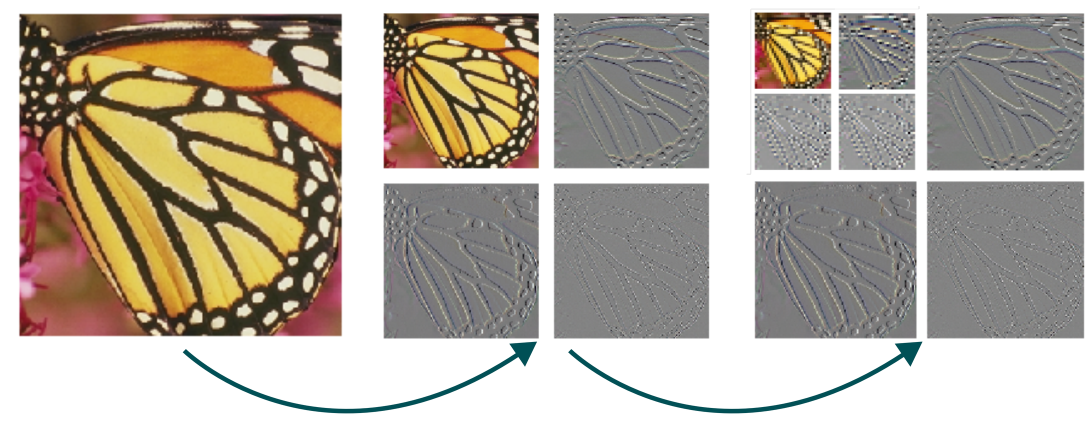
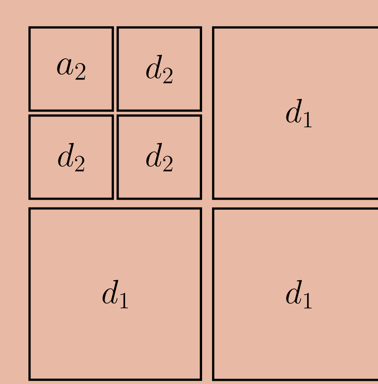


Image as wavelet blocks

Orthonormal wavelet transform: can be performed *recursively* up to a scale J to get a decomposition

$$x = W^\top \underbrace{(a_J, d_J, \ldots, d_1)}_{\omega}$$



$J = 2$

Block-coordinate descent (BCD) algorithms

Recent advances

Optimization problem written in wavelet domain [1]

$$\hat{x} = W^\top\hat{\omega} \text{ with } \quad \hat{\omega} \in \underset{\omega=(a_J,d_J,\ldots,d_1)}{\text{argmin}} \left[\underbrace{\frac{1}{2}\|AW^\top(a_J, d_J, \ldots, d_1) - y\|_2^2}_{f(a_J,d_J,\ldots,d_1)} + \underbrace{\|\Lambda(a_J, d_J, \ldots, d_1)\|_1}_{\lambda_{a_J}\|a_J\|_1 + \sum_j \lambda_{d_j}\|d_j\|_1} \right]$$

Block-coordinate descent Forward-Backward algorithm

Given: Step size τ, regularization parameter λ
Input: Initial coefficients $(a_J^0, d_J^0, \ldots, d_1^0)$
for $k = 0, 1, 2, \ldots$ **do**
  $a_J^{k+1} = a_J^k + \varepsilon_{a_J^k}\left(\text{prox}_{\tau\lambda_{a_J}\|\cdot\|_1}(a_J^k - \tau\nabla_{a_J} f(a_J, d_J, \ldots, d_1)) - a_J^k\right)$
  for $j = J, J-1, \ldots, 1$ **do**
    $d_j^{k+1} = d_j^k + \varepsilon_{d_j^k}\left(\text{prox}_{\tau\lambda_{d_j}\|\cdot\|_1}(d_j^k - \tau\nabla_{d_j} f(a_J, d_J, \ldots, d_1)) - d_J^k\right)$
  end for
end for

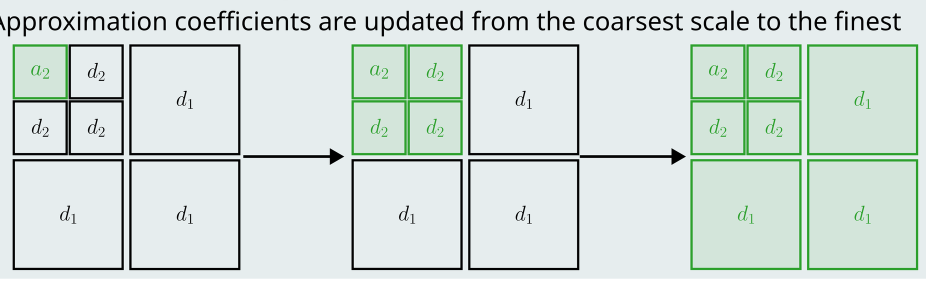
$(\varepsilon_{a_J^k}, \varepsilon_{d_J^k}, \ldots, \varepsilon_{d_1^k}) \in \{0,1\}^{J+1}$ allows to choose which blocks are updated at each iteration

$\varepsilon_{b^k} = 1$: block b is updated at iteration k

$\varepsilon_{b^k} = 0$: block b is not updated at iteration k

Coarse-to-fine update rule (MLFB, [1])

Approximation coefficients are updated from the coarsest scale to the finest



The performance depends on the context:

- **High blur / low noise:** accelerates convergence
- **Low blur / high noise:** performs worse than methods that are better distributed across frequencies

Challenge: define a rule that *automatically adapts* to the degradation regime.

MAGIC-FB

Contribution

Idea: Dynamically sample the *most impactful blocks* at each iteration, drawing inspiration from a *proximal Gauss-Southwell rule.*

Method: Given the current wavelet coefficients $(a_J^k, d_J^k, \ldots, d_1^k) =: (w_0^k, \ldots, w_J^k)$, define the *block-wise proximal update residual*

$$\Delta_i^k := w_i^k - \text{prox}_{\tau\lambda_i\|\cdot\|_1}\left(w_i^k - \tau\nabla_{w_i} f(w^k)\right)$$

Define activation probabilites

$$p_i^k = \frac{\|\Delta_i^k\|}{\|\Delta^k\|} \in [0,1]$$

Activate block i according to

$$\varepsilon_i^k \sim \mathcal{B}(p_i^k)$$

Computational efficiency: the linearity of the problem allows for cheap incremental updates of the gradients

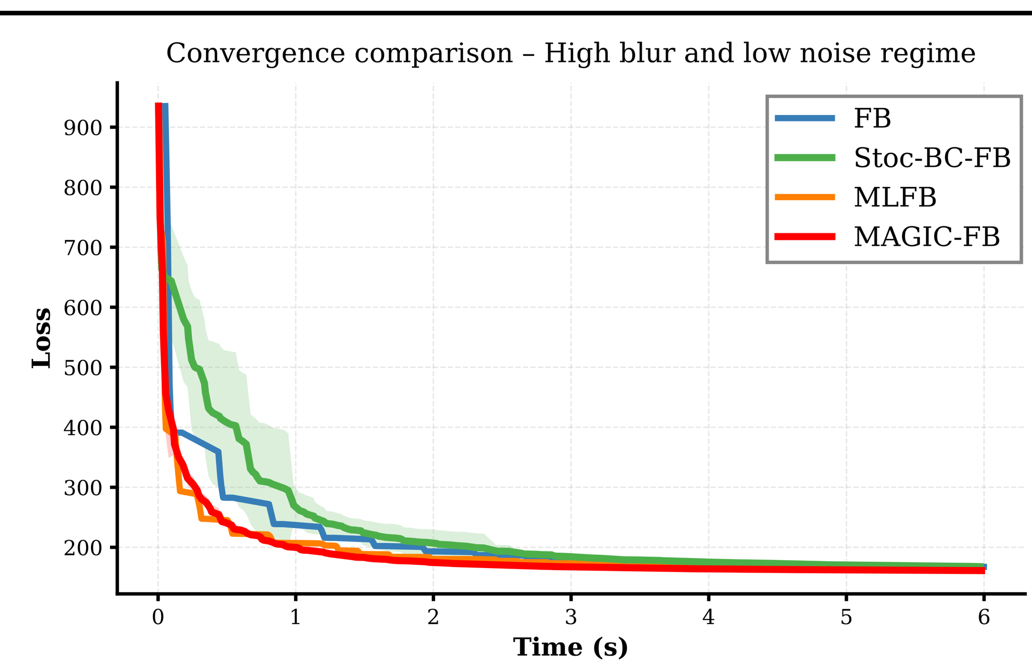
$$\nabla_{w_i} f(w) = \sum_{j=0}^{J} \Pi_i A^\top A \Pi_j^\top w_j - \Pi_i A^\top y$$

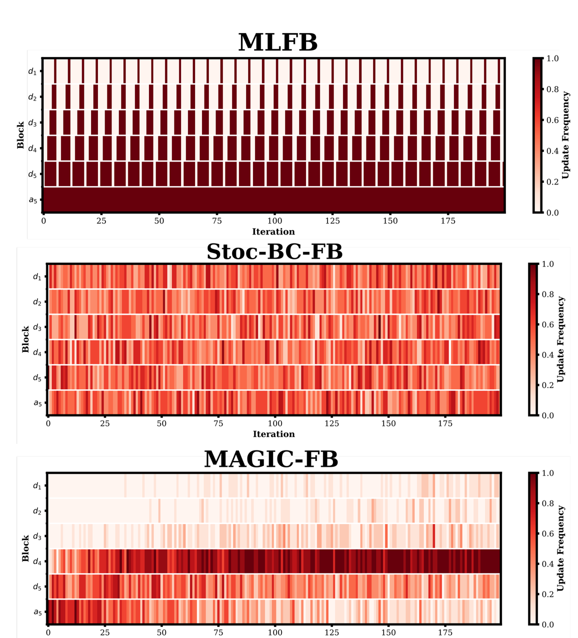
When block i^k is updated, we only need to recompute the $J+1$ terms

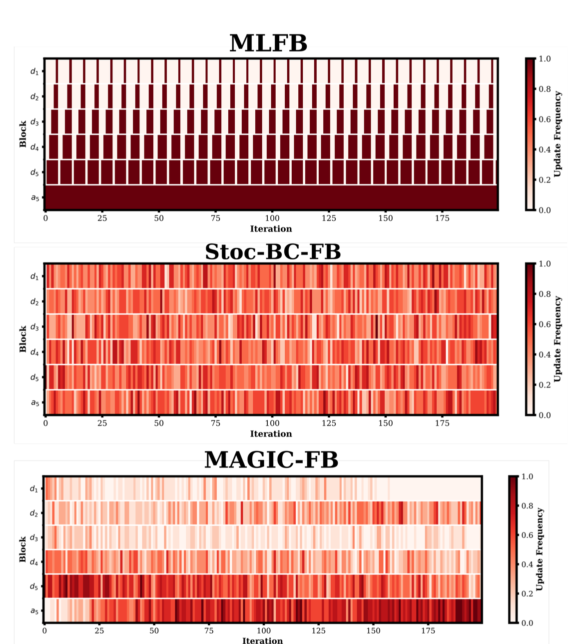
$$(\Pi_i A^\top A \Pi_{i^k}^\top w_{i^k}^{k+1})_i$$

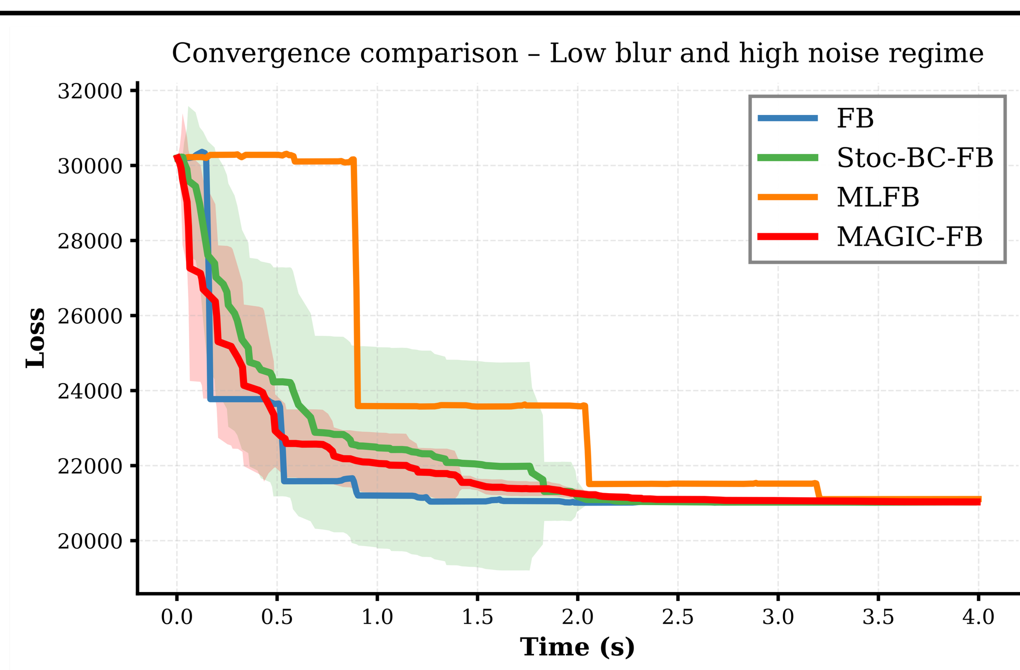
Convergence guarantees: convergence results in expectation can by derived from [2] under the assumption that there exists $p_{\min} > 0$ such that

$$\forall 0 \le i \le J, \forall k \in \mathbb{N}, \quad p_i^k \ge p_{\min} > 0$$

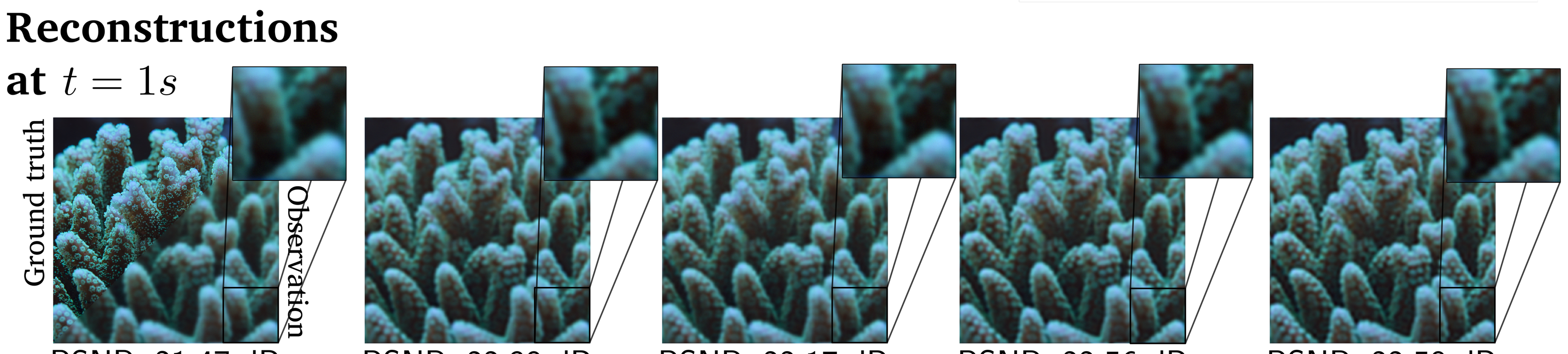








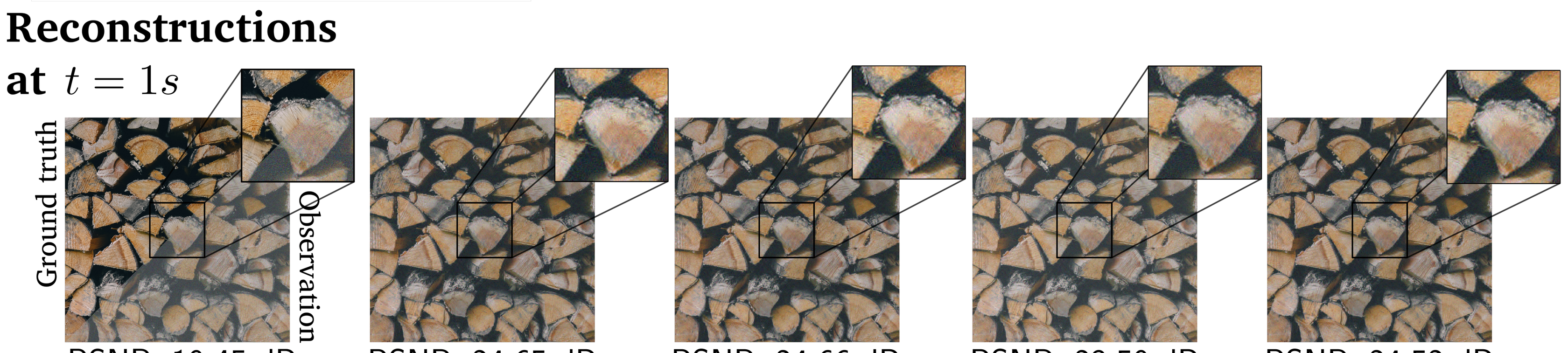
Reconstructions at $t = 1s$



PSNR: 21.47 dB | PSNR: 22.38 dB | PSNR: 22.17 dB | PSNR: 22.56 dB | PSNR: 22.58 dB

FB | Stoc-BC-FB | MLFB | MAGIC-FB

Reconstructions at $t = 1s$



PSNR: 19.45 dB | PSNR: 24.65 dB | PSNR: 24.66 dB | PSNR: 23.59 dB | PSNR: 24.52 dB

FB | Stoc-BC-FB | MLFB | MAGIC-FB

Perspectives:

- **Plug & Play extension:** to improve the quality of the reconstruction, replace the proximal operators by a learned denoisers
- **Unrolled extension:** instead of a fixed wavelet transform, learn a decomposition into blocks to study the structures that naturally stand out in imaging problems
- **Convergence rates w.r.t. the probabilities:** derive convergences rates that depend on the selection rule to theoretically show which method is the fastest

[1] Briceño-Arias, L., Gonçalves, P., Lauga, G., Pustelnik, N., Riccietti, E. (2025). *A flexible block-coordinate forward-backward algorithm for non-smooth and non-convex optimization.* arXiv preprint.

[2] Salzo, S., Villa, S. (2022). *Parallel random block-coordinate forward-backward algorithm: a unified convergence analysis.* Mathematical Programming, Springer.